Computer Science 605.411

Problem Set 2 Name:_____

1. Use hexadecimal notation to show the result produced by each of the following instructions (both of which are described in appendix B) and indicate where the result would be stored. Prior to executing each instruction, registers $8 and $9 are reset to contain the two's complement representation of the negative decimal value -2147483648 (in $8) and the two's complement representation of -4 (in $9):

(5) a) addu $8,$9,$8 _____

(5) b) xor $9,$8,$9 _____

2. (10) a) If the 32-bit pattern 0xAFFFFFFF is interpreted as an excess-2147483648 integer, what decimal value would it represent?

(5) b) What is the smallest (i.e. the most negative) decimal value that can be represented using a 32-bit excess-65540 system.

(5) c) The value -128 is to be represented using an 8-bit system. For each of the following systems, indicate (yes or no) whether it is capable of representing this value:

I. an excess-128 system
II. a one's complement system
III. a sign and magnitude system
IV. a two's complement system

3. (10) The 64-bit decimal integer value 3904679375210100017 is stored in memory with the most significant byte at address 0x10080000, the next most significant byte at 0x10080001, etc . What ascii character would be represented by the contents of each of the eight bytes?

Byte address	ASCII character represented by contents
0x10080000	
0x10080001	
0x10080002	
0x10080003	
0x10080004	
0x10080005	
0x10080006	
0x10080007	

4. (20)For each of the items listed below, enter a checkmark in the appropriate column to indicate whether the item is a feature of the machine's architecture or a feature of its organization.

Item	Architectural feature	organizational feature
Cache memory		
Multiplexed I/O bus		
Indexed addressing mode		
128-bit CPU registers		
Flash based storage		
Vector instructions		
Micro-programmed control		
Zero-address instructions		
USB serial I/O port		

5. The two's complement integer representation of the decimal value -134217728 is contained in register $7 and the IEEE standard 32-bit floating point representation of the same value is contained in register $f4. Without making any assumptions about the contents of any other registers or of any memory words,
(10) a) show a MIPS assembly language instruction sequence, containing only **True-ops**, that would compute the integer sum of $7 plus the decimal constant 15 and leave the result in $7. Recall that **True-op** instructions are native built-in instructions that the hardware understands as opposed to pseudo-instructions (also called synthetic instructions) for which the assembler must substitute one or more built-in instructions.

(10) b) What would be the decimal equivalent of the resulting pattern left in $7 by this instruction sequence?

(10) c) Using only registers, immediate operands and no memory variables, show an instruction sequence consisting only of MIPS **True-op** instructions that would compute the floating point sum of $f4 plus the floating point value 15.0 and leave the result in $f4. Make no assumptions about the contents of any other registers. Appendix B of the text lists all instructions and indicates whether each is a true-op or a pseudo-instruction.

(5) d) What floating point value would be represented by the bit pattern left in $f4 if rounding IS NOT performed (i.e. the result is simply truncated to fit into the available bits). Express your answer as a decimal value.

(5) e) What floating point value would be represented by the bit pattern left in $f4 if rounding IS performed? Express your answer as a decimal value.